KARACRIX 入門実用ガイド

1章　基礎知識

（章別取扱説明書 v3.00）

株式会社 エスアイ創房

■改定履歴

第 3.00 版　2009/11/01

■ おことわり

(1) 本書内容の一部又は全部を、無断で他に転載することは禁止されています。

(2) 本書内容は、将来予告無く変更する場合があります。

KARACRIX は株式会社エスアイ創房の登録商標です。

Microsoft,Windows,Excel は米国 Microsoft Corporation の登録商標です。

その他、本文中に記載されている社名および商品名は、一般に開発メーカーの登録商標です。

KARACRIX 入門実用ガイド 第 3.00 版　© S.I.Soubou Inc.

目次

1 章　基礎知識 1-5
1.1 計測制御の知識 1-5
1.2 制御とは 1-5
1.3 計測とは 1-3
1.4 計測制御システムとは 1-4
1.5 センサとアクチュエータの知識 1-5
1.6 オブジェクトとは 1-7
1.7 センサとアクチュエータをつなぐリモート I/O 装置について 1-9
1.8 センサとアクチュエータをリモート I/O 装置につなぐ（配線例） 1-10
1.9 リモート I/O 装置と PC をケーブルでつなぐ 1-11
1.10 リモート I/O 装置と PC をプログラムでつなぐ 1-12
1.11 ユーザプログラムと通信ドライバ(S1)の協調 1-13
1.12 ユーザプログラムの作成と実行 1-14
1.13 ユーザプログラミング簡単入門 1-15
1.14 C プログラムによる「Hello World」画面表示 1-17
1.15 リレーの操作 1-18
1.16 KaracrixBuilder＋リモート I/O 装置で実現できること 1-19
1.17 付録 1-21

1章　基礎知識

1.1　計測制御の知識

今日に於いては、日常生活の様々な所で、コンピュータによる制御システムが稼動しています。毎日の通勤通学で通る駅の自動改札やビルのエレベータ、自宅のエアコンや洗濯機など、私達の生活を便利にしてくれる機能はほとんどがコンピュータによる制御システムによって実現されているのです。
ここでは、KarcrixBuilder でこのような様々な制御システムを実現する為の基本的な知識に触れておきます。

1.2　制御とは

制御とは、「目標に向かって必要な操作を加えて行くこと」と定義されています。この目標とは、例えば「指定された階でエレベータを止める」ことだったり、「洗濯が終ったら、次はすすぎをする」ことであったりとシステムによって様々なケースがあるでしょう。そして、ある時点の目標に対して必要な操作を行う事により、目標値に適合するよう機器をコントロール（調節）する事、と言えます。

現在よく使われる制御の方法は、大きく分けて次の2つに分類できます。
・シーケンス制御　(順序制御)
・フィードバック制御　(目標値制御)

シーケンス制御は、決められた手順に従って次々と制御を進めて行くやり方で、順序制御とも言われています。
洗濯機を例に取れば、

1) 水を適量まで入れる
2) 洗濯をする
3) すすぎをする
4) 脱水をする

と言うように、ある作業が終了したら次の作業を行う様に、決められた手順を正確に実行する方法です。

次に、フィードバック制御は、目標値制御とも言われており、ある目標値を達成する為に制御量を調節するやり方です。
先きほど述べた洗濯機を例に取れば、

1) 水を適量まで入れる

この手順の作業の詳細は、フィードバック制御で実現されています。これは、「水が適量になった」という目標値を達成する為に給水を続けるかどうかを判断して目標を達成したら給水を終了するという制御を行っているのです。

以上の事から分かるように、シーケンス制御とフィードバック制御は一つの制御システムの中で組み合わされて使用されている場合が多く、この2つの制御を組み合わせる事で、ほとんど全ての制御システムを組む事が可能となります。

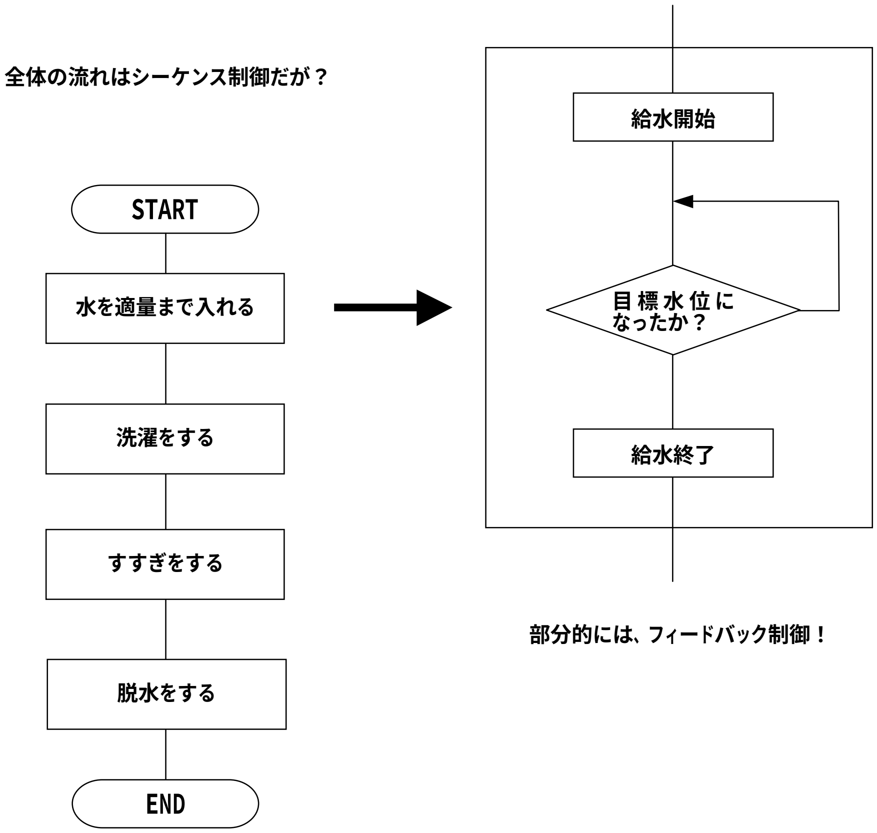


図 1.2.1 シーケンス制御とフィードバック制御

1.3 計測とは

計測とは、「事象を量的に捕らえてその値を使用する事」だと言えます。 事象を量的に捕らえる為の装置をセンサと呼んでいます。 センサは、物理量(熱、力、仕事など)や化学量(濃度など)などをコンピュータで処理しやすい電気信号(電圧、電流、抵抗など)に変換します。

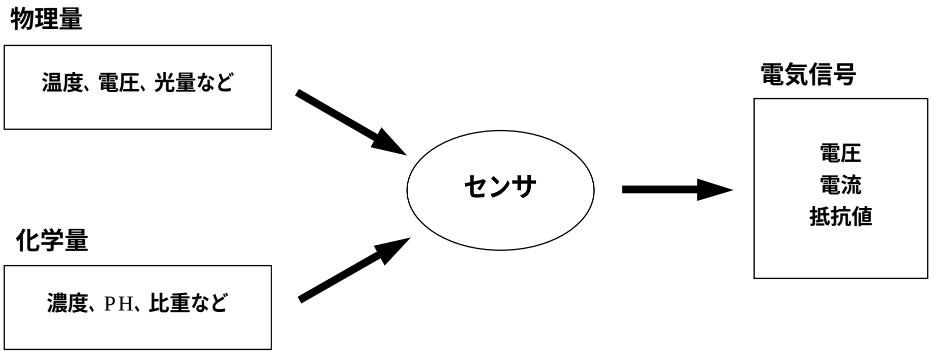


図 1.3.1 物理量を電気信号へ

センサが出力する信号には、時間の経過に伴って連続的に変化するアナログ(連続値)形式のものと、設定値を越えると ON/OFF(入/切)の 2 つの状態変化を検出するデジタル(2値)形式のものがあります。

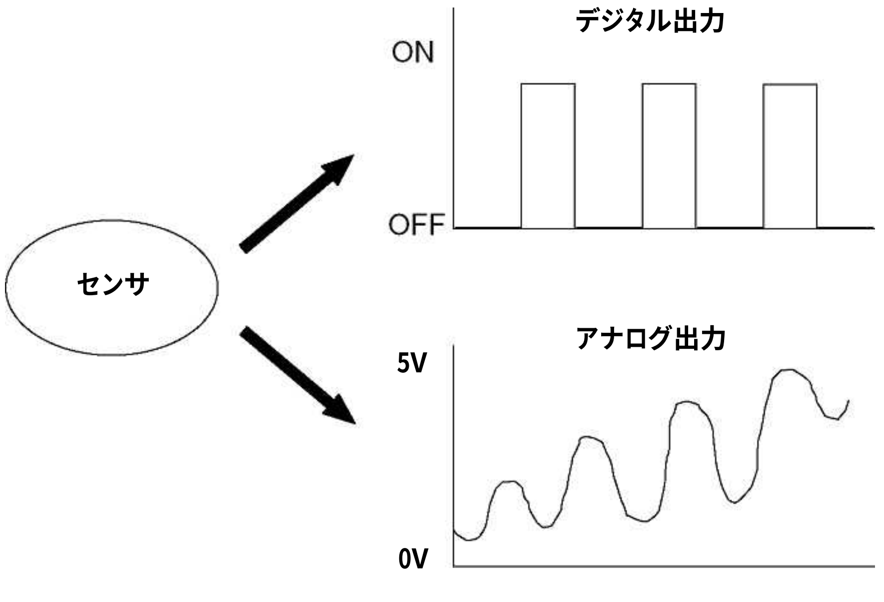


図 1.3.2 アナログとデジタル

現代に於いて計測とは、センサからの電気信号をコンピュータに取り込んで処理する事が一般的になっています。 また、計測したデータ自体に意味がある場合もありますが、制御システムを構成する一機能として計測を捉える事も多くなっています。

1.4　計測制御システムとは

実際には、計測と制御は一体となってシステムを構成する事になりますので計測制御システムと言うのが正確な表現です。
計測制御システムをコンピュータで構築する為には、

1) 計測値を数値としてコンピュータに取り込む仕組み
2) コンピュータで処理した制御情報を制御対象に反映する仕組み

が必要となります。これを実現する為の機能を持っているものが、次項で述べるセンサとアクチュエータです。

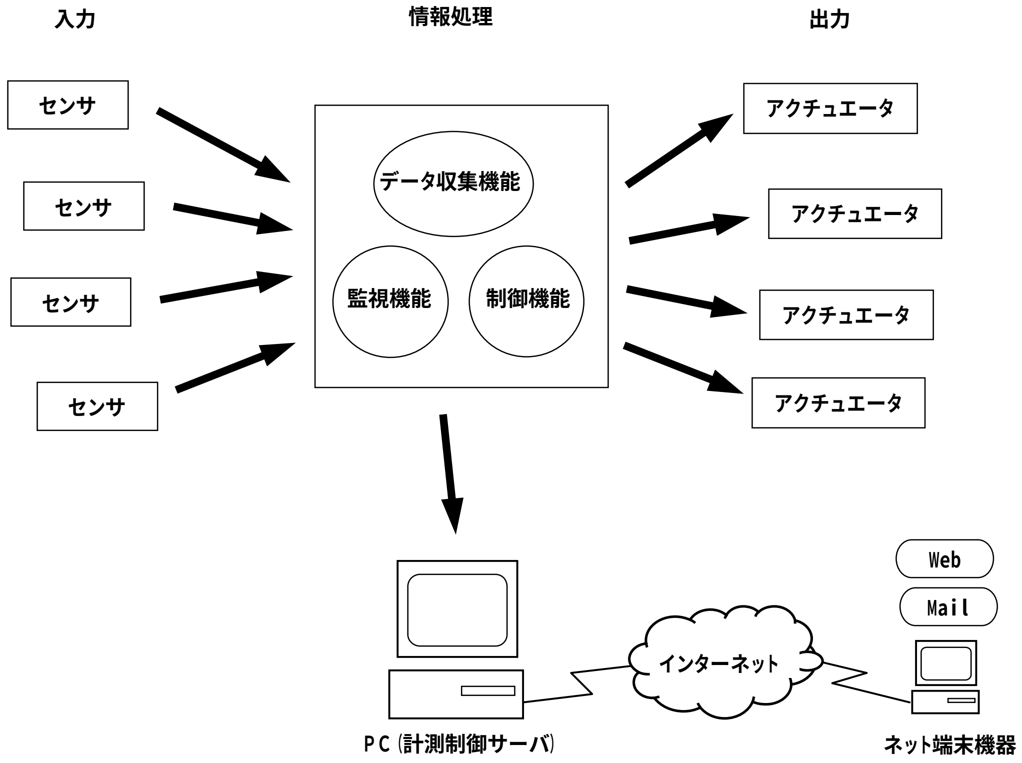


図 1.4.1 計測制御システム図

1.5　センサとアクチュエータの知識

● センサとは

センサとは計測対象の状態を検出するものです。
センサには、大きく分けて2つのタイプが使われています。
1) 導体の開閉(ON/OFF) または 2値電気信号を出力するタイプ　(デジタル(2値)タイプ)
2) 検出値を連続的に変化する電気信号にして出力するタイプ　(アナログ(連続値)タイプ)

1) デジタルタイプ

デジタルタイプは、物体のON/OFF状態をスイッチの開閉(導通、不通)として、または電圧電流抵抗などを2値の高低差(TTLの0/5vなど)にして出力するものがあります。このデジタル信号は計量の目的でカウント値として使用する場合もあります。

写真 1.5.1 デジタルセンサ (スナップスイッチ、フロート(水位)スイッチ、近接スイッチ、傾きスイッチ)

2) アナログタイプ

アナログタイプは、温度、電力、圧力などのアナログ情報を電気信号に変換して出力するものです。
一般的にアナログタイプのセンサそのものから出力される電気信号は微小なため、扱いやすい電圧電流レベルに(センサ内部で増幅)変換して出力されるものもあります。この場合の出力としては 0～1V,0～5V,0～10Vなどがあります。

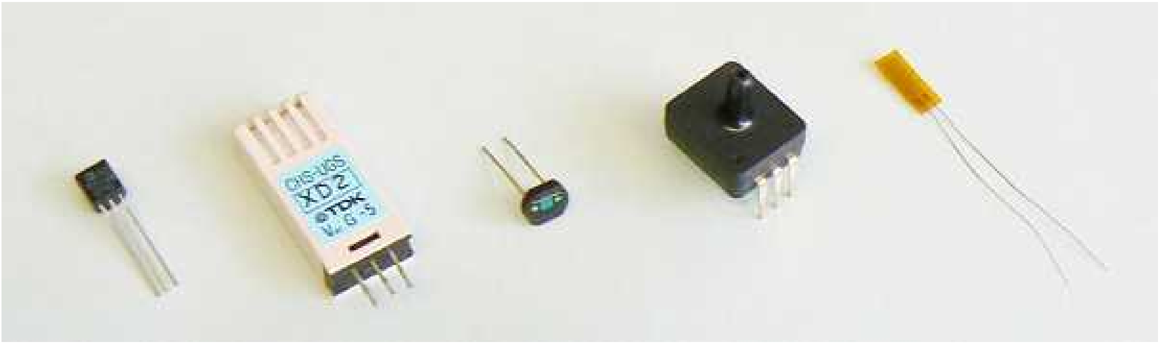


写真 1.5.2 アナログセンサ (温度、湿度、照度、圧力、ひずみ)

● アクチュエータとは

アクチュエータとは制御対象の物を動かすものです。
アクチュエータは、電力を受けて物理動作(回転運動、直線運動、音響出力など)に変換します。
この電力の電圧には、12V、24V、48V、100V 等の直流または交流がよく使われています。 アクチュエータとして実際に使われるものに、モーター(ポンプ、ファン、ロボットアーム等)、ソレノイド(電磁弁、電磁リレー等)、警報器(ベル、ランプ等)などがあります。

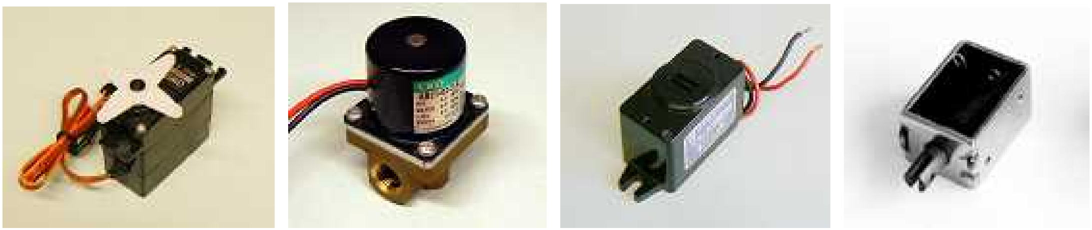


写真 1.5.3 アクチュエータ(サーボモータ、電磁バルブ、圧電ブザー、ソレノイド)

なお、電気操作信号 (微小電力) からアクチュエータを直接駆動させることが出来ない場合には、電力制御回路を仲介させる必要があります。

図 1.5.1 制御信号の流れ

具体的には、以下の図に示すようなトランジスタあるいは小型リレーからパワーリレーを駆動させて大きな接点出力を得るための電力供給回路を仲介させます。

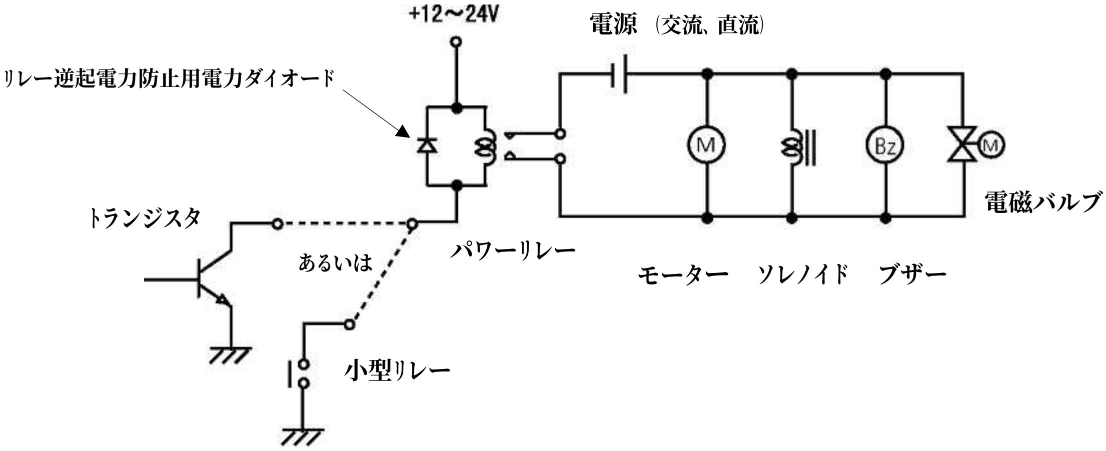


図 1.5.2 電力制御回路 (パワーリレー使用例)

1.6 オブジェクトとは

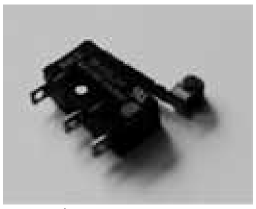
KaracrixBuilder では、センサやアクチュエータなど入出力対象のことを総称してポイント型のオブジェクト(対象物)※と呼んでいます。そして、これらオブジェクトの状態情報は、KaracrixBuilder のオブジェクト用に割り当てられたメモリに取り込まれシステム上で共有して使用されます。
オブジェクトには、実ポイントオブジェクトと擬似ポイントオブジェクトという概念があります。センサやアクチュエータと直接ひも付いているポイントは実ポイントオブジェクトという分類で扱います。演算結果や目標値など、実体(センサなど)と直接ひも付かないものは擬似ポイントオブジェクトとして扱います。

(詳しくは、KaracrixBuilderV3 システムマニュアルを参照してください)

※ソフトウェア技術の1つにオブジェクト指向と言うものがあります。これは KaracrixBuilder で使用するオブジェクトの概念と異なりますので解釈にご注意下さい。

KaracrixBuilder で扱う実ポイントオブジェクトのセンサとアクチュエータ機器例を下表に示します。

表 1.6.1 センサとアクチュエータとその対応オブジェクト

<table>
<tr><th></th><th colspan="2">機器</th><th>信号形態</th><th colspan="2">オブジェクト型</th></tr>
<tr><td rowspan="4">入力（センサ）</td><td rowspan="2">デジタル</td><td rowspan="2">
スイッチ

リードスイッチ</td><td rowspan="2">ON/OFF
カウント(計量)</td><td>DI</td><td>Digital Input</td></tr>
<tr><td>PI</td><td>Pulse count Input</td></tr>
<tr><td rowspan="2">アナログ</td><td rowspan="2">
湿度センサ

温度センサ</td><td rowspan="2">電圧
電流
抵抗</td><td rowspan="2">AI</td><td rowspan="2">Analog Input</td></tr>
<tr></tr>
</table>

出力（アクチュエータ）	デジタル	ブザー DC モータ	ON/OFF	DO	Digital Output
	アナログ	メータ	電圧	AO	Analog Output
	PWM	サーボモータ	パルス幅	AO	Analog Output

※センサ、アクチュエータ機器は、一例を示します。

本実用ガイドで用いられる用語、「ポイント」「ポイントオブジェクト」は、同義語です。
また、「ポイント ID」「オブジェクト ID」「ポイントオブジェクト ID」「OBJID」も全て同義語です。

1.7　センサとアクチュエータをつなぐリモート I/O 装置について

リモート I/O 装置とは、通信機能を有した I/O 装置のことです。またこの I/O 装置とは、センサから出力される電気信号を装置内に取り込んだり、装置外のアクチュエータに電気操作を行ったりする信号の入出力 (Input/Output) 装置です。センサの入力信号やアクチュエータへの操作信号を数値データ等に変換して、かつ決められた通信ルールに従って外部(PC)とデータ接続します。

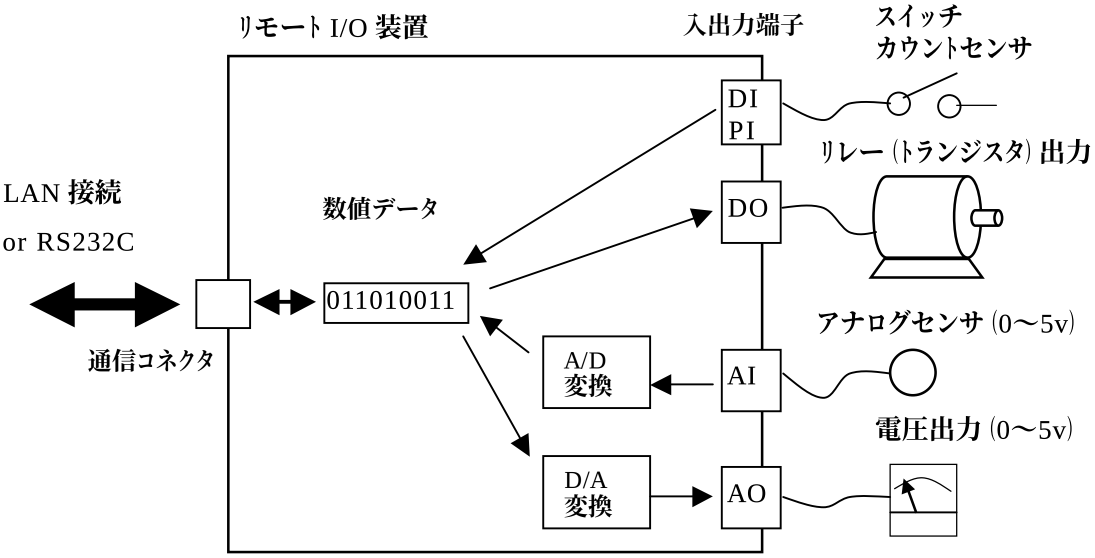


図 1.7.1 リモートI/O装置の機能概要(TK0040A)

装置は、デジタル入力端子 (DI,PI) 、デジタル出力端子 (DO) 、アナログ入力端子 (AI) 、アナログ出力端子 (AO) を搭載しており各端子には以下のような機器を接続して使用します。

デジタル入力端子 (DI,PI)　→　スイッチやカウント型センサ　(ON/OFF の状態値、併せてカウント値も取得します)
デジタル出力端子 (DO)　→　リレーやトランジスタ出力　(アクチュエータを ON/OFF します)
アナログ入力端子 (AI)　→　アナログセンサ　(温度や電力などを電圧値で取得します)
アナログ出力端子 (AO)　→　電圧出力　(指定したレベルの電圧を出力します)

1.8　センサとアクチュエータをリモート I/O 装置につなぐ（配線例）

●DI、PI デジタル入力：スイッチの場合

装置内のフォトダイオードが点灯するようにスイッチ電流を流します。点灯すると装置は ON を検出します。

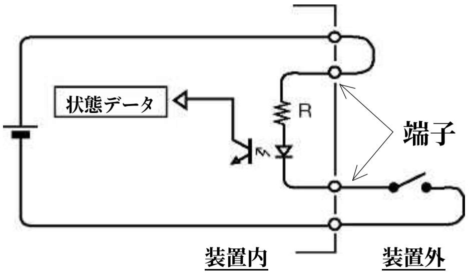


図 1.8.1 デジタル入力回路

●DO デジタル出力：リレー及びトランジスタの場合

リレーやトランジスタを短絡 (ON) させることにより、負荷電流を流し接続機器を ON とさせます。

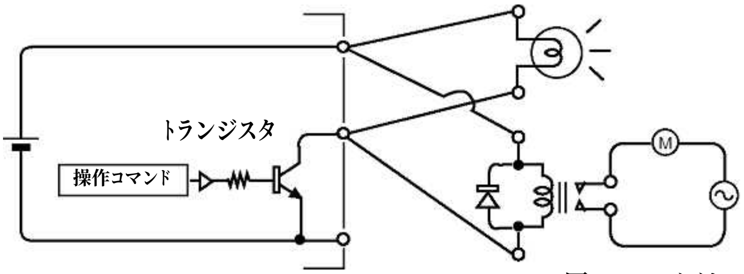


図 1.8.2 デジタル出力回路 (トランジスタ)

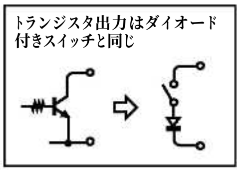


図 1.8.3 トランジスタ出力

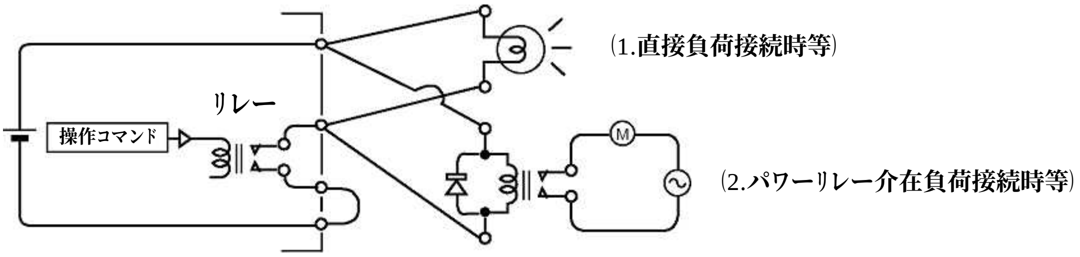


図 1.8.4 デジタル出力回路 (リレー)

●AI アナログ入力：電圧型センサの場合

装置内の A/D コンバータを使用してアナログ入力信号を数値データに変換します。コンバータには、10、12、16 ビット分解能のものが良く使われ分解能（精度）はそれぞれ 1/1023(0.1%)、1/4095(0.025%)、1/65534(0.0015%) になります。(TK0040A は 10 ビット分解能です)

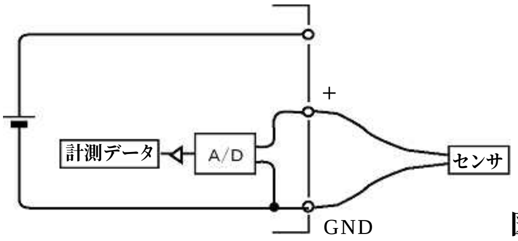
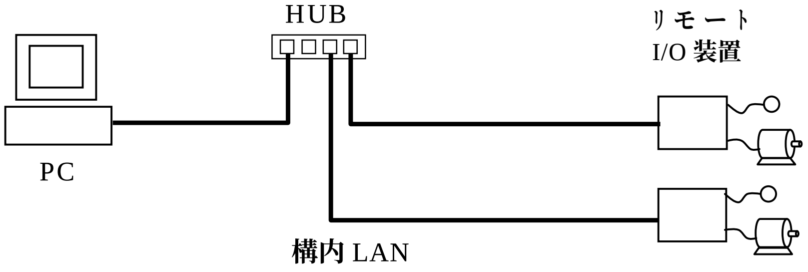


図 1.8.5 アナログ入力回路

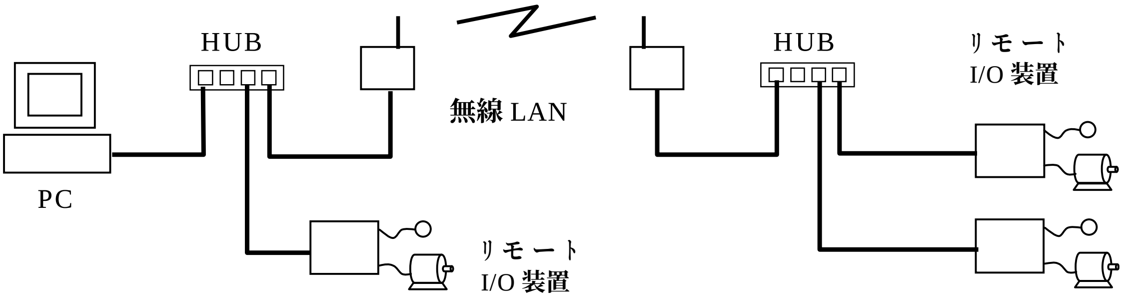
1.9　リモート I/O 装置と PC をケーブルでつなぐ

リモート I/O 装置とPCは、LAN や RS232C などの通信ケーブルを使用して接続します。特に、LAN ケーブル(IP 通信)を使用する事により、広域に分散したセンサやアクチュエータを繋ぐことが可能となります。以下に、LAN を使用した時の各接続例を示します。

(構内 LAN 接続)



(無線 LAN 接続)



(広域 LAN 接続)

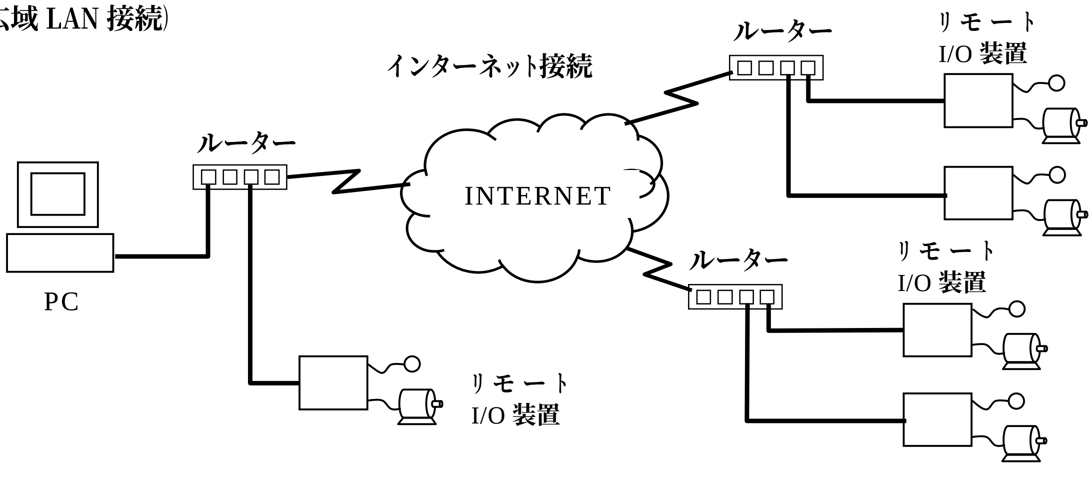


図 1.9.1 LAN 接続型リモート I/O 装置の各接続例

1.10 リモート I/O 装置と PC をプログラムでつなぐ

リモートI/O 装置は、通常※PC から指示を受けて動作するようになっています。PC からリモートI/O 装置に指示を与えるには、データ通信をする必要があります。この通信方式には色々な方式がありますが、ポーリングアクセス方式というものが手軽です。この方式は、PC からリモート I/O 装置に対して処理目的に合ったコマンド(指示)を送り、リモートI/O 装置がこれに応える(アンサー)というものです。
PC からリモートI/O 装置に対して「計測値読み込みコマンド」を送ると、リモートI/O 装置は PC に計測データを返します。また、例えばリレーON 操作コマンドを送ると、リモートI/O 装置はリレーを ON させ結果を返します。
この様に、コマンドと応答を用いてリモートI/O 装置と PC をデータ接続します。

※リモートI/O 装置にはイベント送信機能と言うものがあり、これを用いた場合に装置は自発能動的に動作します。

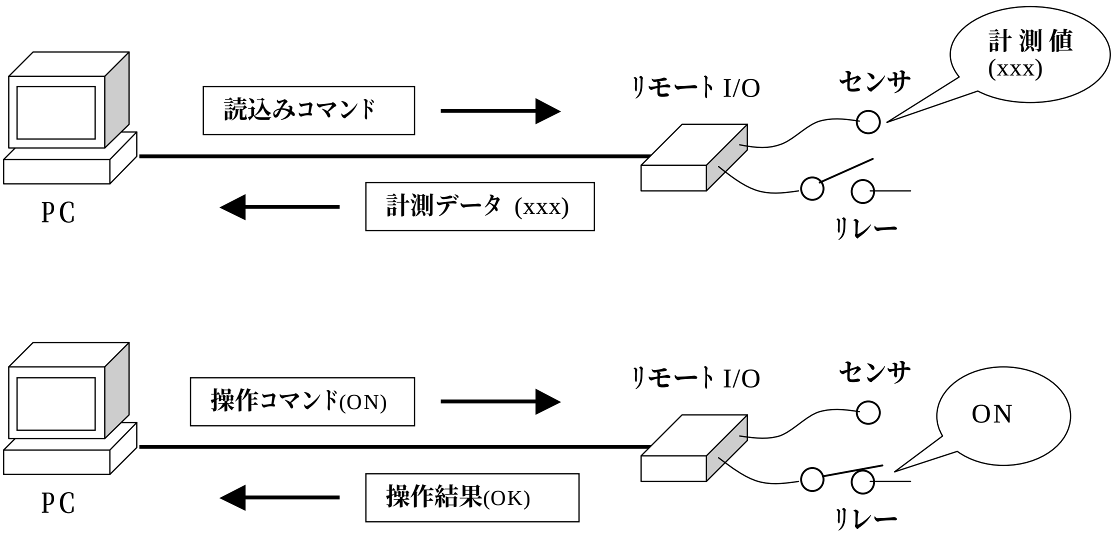


図 1.10.1 リモートI/O 装置と PC の通信

1.11 ユーザプログラムと通信ドライバ(S1)の協調

リモートI/O装置に対してコマンドを送信して応答データを受信する手順には、人間の会話と同様にルールというものがあります。このルールのことを**プロトコル**と呼んでいます。リモートI/O装置を使うには、このプロトコルに対応した通信プログラム(通信制御ドライバ)を作成する必要があります。

自動制御プログラムを作成する場合機能的に大まかに分けますと、取得したセンサ等のデータを処理して利用するユーザアプリケーション部と、リモートI/O装置と通信してデータを取得してくる通信部を作成する必要があります。ユーザアプリケーション部分の処理が小さい場合は「ユーザアプリケーション機能＋通信機能」を1本にまとめて作成しても良いですが、アプリケーション部分が多機能になる場合は「ユーザアプリケーション機能」と「通信機能」を別々に構成したほうが見通しの良いプログラム構成にすることができます。

そこで、弊社ではこの通信機能を集約させておおよそ共通に使える「**通信制御ドライバ S1**」(以下、S1ドライバ) ※というプログラムを提供しています。本ガイドでは、この S1 ドライバの利用を前提に解説しています。S1ドライバを使用してシステムを構成する場合は、ユーザプログラム＋S1ドライバによるマルチプロセス構成になります。この場合、ユーザが作成するプログラムには、通信機能を組み込む必要はありません。機器操作制御部分のアプリケーションプログラムのみを書けば良いと言う事になります。

※通信制御ドライバ S1 はプログラム内容が公開されているものです。このプログラムを参考にご自身の利用に最も適したものに再作成して使用されても構いません。

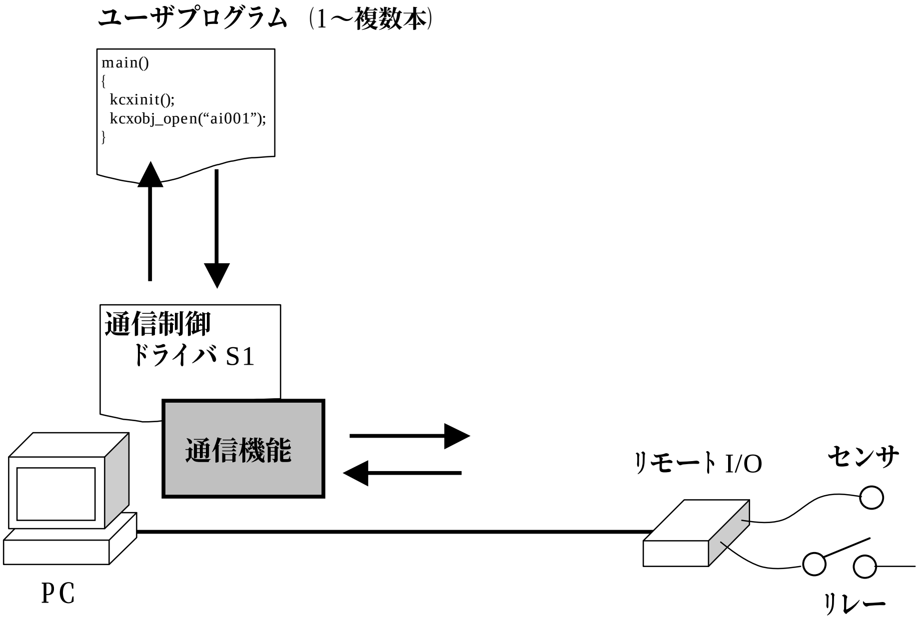


図 1.11.1 通信制御ドライバ S1 が通信機能を担当

1.12 ユーザプログラムの作成と実行

KaracrixBuilder では、C 言語という文法でユーザプログラムを記述する必要があります。
プログラムの実行までの手順は、以下のようになります。
①ユーザプログラムを記述します
②作成したプログラムをコンパイル(翻訳)と言って PC で実行できる実行形式(機械語)に変換します。
③変換された実行形式のプログラムを PC で実行します。

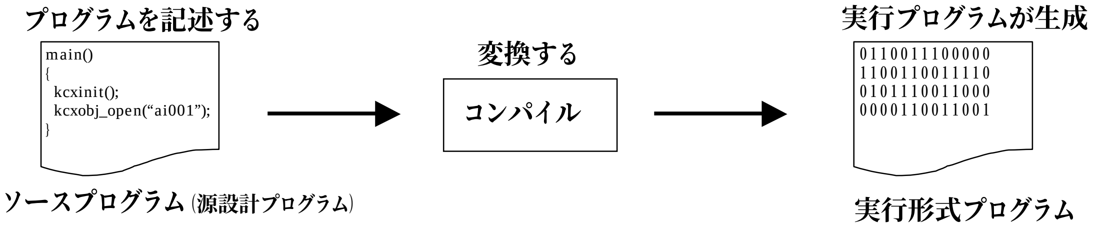


図 1.12.1 実行プログラムの生成

実行するプログラムは、ユーザプログラムと通信制御ドライバ S1 が同時に実行されるように手動実行あるいは自動起動設定(プログラムオブジェクトを起動するプログラムを準備、あるいはスケジュール起動に設定)しておきます。この組み合わせでは、リモート I/O 装置へ操作コマンドを送ったり応答データを受け取ったりする機能を通信制御ドライバ S1 が担い、その上位に当たるアプリケーション的な操作指示をユーザプログラムが担い協調動作させます。

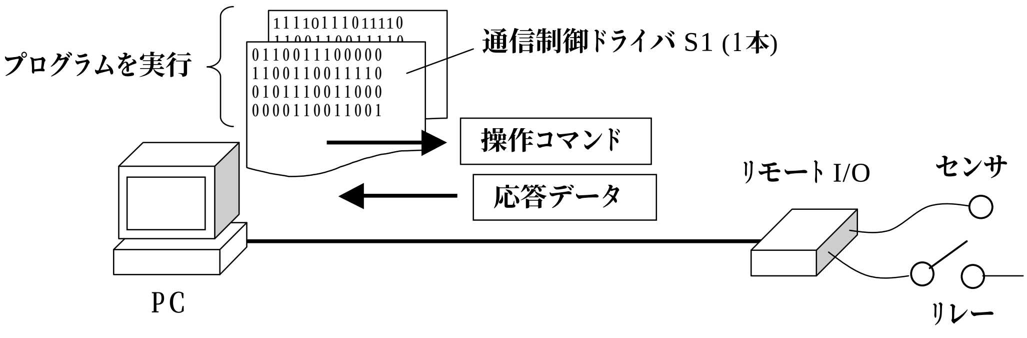


図 1.12.2 プログラムからリモート I/O 装置へ指示

1.13 ユーザプログラミング簡単入門

KaracrixBuilder を操る C 言語の文法を知らない方の為に、C 言語を使用した最小のプログラム構成を以下に示しますので学習してみてください。ご覧のように、include 文と main 関数(Function)で構成されているシンプルなものですが、これだけの記述で C 言語のプログラムとしては完成しています。

```
#include   <stdio.h>
main( int argc, char *argv[] )
{
    ※A
}
```

但し、上記のプログラムは完成形ではありますが、実行しても何も起きません。それは、処理させたい実行文が記述されていない為です。実行文は、main 関数内の { と } に挟まれた※A のところに記述していきます。

次に、KaracrixBuilder 上でプログラムを記述する際に2つの記述ルールがあります。

①include 文には karacrix.h ヘッダファイルを指定する

一般的な C 言語では include 文に stdio.h (ヘッダファイル) 等を指定しますが、KaracrixBuilder ではこの代わりに karacrix.h を指定します。stdio.h 等の各種ヘッダファイル指定を複数行記述しなくても良いように必要なものを karacrix.h に集約しています。また karacrix.h には、KaracrixBuilder システム固有で必要なものも内包されています。(詳しくは、KaracricxBuilderV3 システムマニュアル参照)

②kcxinit 関数を先頭に置く

main 関数内の最先頭の実行文に kcxinit(argc,argv); という関数を記述してください。これは、KaracrixBuilder の初期化に必須なものとなります。(詳しくは KaracricxBuilderV3 システムマニュアル参照)

以下のプログラムは、上記プログラムを KaracrixBuilder 用に修正したものを示します。 実行文は、kcxinit(argc,argv); 文以降から } に挟まれた※B のところに記述していきます。

```
#include   <karacrix.h>
main( int argc, char *argv[] )
{
   kcxinit(argc,argv);
   ※B
}
```

● C 言語概要

C 言語の仕様について概要をまとめてみましたので参考にして下さい。詳しい言語仕様については他の文法書を参照して下さい。

○記述文法

1. プログラムには、実行の始まりを示す main 関数が必要です。
2. プログラムは、関数の集まりによって構成されます。(main 関数1つでも構いません)
3. 関数内の実行文は { と } の間に記述されます。
4. 実行文の終わりにセミコロン（ ; ）を付けて区切ります。
5. 変数名は英小文字で記述するのが慣例で、強調させたい場合に大文字を使用します。
6. 変数名は、大文字小文字区別されるので注意が必要です。(abc と Abc は別ものです)
7. 変数名の先頭は、英字（a～z、A～Z）あるいは下線（ _ ）でなければなりません。
8. 変数名に、予約語(if,for 等)を使用してはいけません。
9. コメント（注釈）は /* と */ で囲みます。

○データには2種類ある

定数：値を変えない固定値です。(例:整数=1, -2,99/実数=3.14/文字="ABC")

変数：可変の値をとりうるメモリです。(変数名例:abc,ondo,volt,Volt)

○変数の型

char(文字型), int(整数型), long(倍長整数型), float(単精度実数型), double(倍精度実数型)

※KaracrixBuilder では、実数に精度の高い倍精度実数型の double を使用することに統一しています。

○演算子

1. 代入演算子（=）(例: a=10; a=b;) (誤例: 10=a;)
2. 算術演算子（+,-,*,/）(例: a= 1 + 2)

○制御文（文の流れを制御するもの）

if, for, while, switch, break, continue, return 等

● C 言語とは

C 言語の特徴は、システム記述言語としての完成度が高くかつ広い汎用性と普及率を有していることです。専門アプリケーションの分野では C 言語以外のプログラム言語が多く使われるようになっていますが、その殆どは C 言語をベースに用途別機能を加えた派生的なものと考えて良いでしょう。

この様な経緯もあり、現在主流のプログラム言語（Java,C++,C#,PHP,Ruby など）の基本となるC言語の習得に無駄は考えにくいです。また、上記よりプログラム記述方法も似ていることから派生プログラム言語への移行もスムースです。なお、現時点においても C 言語は最高性能を追求するＯＳの記述言語としては不動の地位にあるようです。

1.14 C プログラムによる「Hello World」画面表示

C 言語入門書などでよく紹介される表示サンプル例が以下のプログラムです。このプログラムを実行した場合、「Hello World.」の文字列が端末画面に出力されます。

```
#include   <stdio.h>
main( int argc, char *argv[] )
{
      printf("Hello World.\n");
}
```

これを、KaracrixBuilder の監視画面あるいは Web 画面等で表示するにはどうしたらよいでしょうか。
ここでは、手順だけを説明しておきますので KaracrixBuilder に慣れた時に試してみてください。
KaracrixBuilder 上で、以下に示すプログラムを記述しコンパイルし実行します。

```
#include   <karacrix.h>
main( int argc, char *argv[] )
{
   int   objid;
   kcxinit(argc,argv);
   objid=kcxobj_open("di001");
   kcxobj_atbut_cwt( objid, 1, "Hello World." );
}
```

上記プログラムの意味は、OBJID 名に di001 と設定されたポイントオブジェクトの文字属性1という内部メモリに「Hello World.」という文字列を書き込む操作を示しています。
この文字列を、監視画面 CAD 上で配置する「属性文字」部品描画で上記のポイントオブジェクトの文字属性1の内容を表示させるように設定すれば「Hello World.」の文字列が監視画面や Web 画面等に表示されるようになります。（上記プログラム次第で動的な文字表現も可能になります）

図 1.14.1 属性文字部品の登録

図 1.14.2 "Hello World"プログラムの実行

1.15 リレーの操作

プログラムからリレーを ON させるにはどうすればよいでしょうか。
以下に示すようなプログラムを記述し、コンパイルし実行します。
リレーを出力としたポイントオブジェクトの OBJID 名を do001 とします。この OBJID をオープンし、これに対して1を送信すればリレーが ON 操作されます。0 を送信すれば OFF 操作されます。なお、本プログラムを実行する場合にも通信制御ドライバ S1 が並列して動作している必要があります。

```
#include    <karacrix.h>
main( int argc, char *argv[] )
{
    int objid;
    kcxinit(argc,argv);
    objid=kcxopen(“do001”);
    kcxobj_sndistat_tokcx(objid, 1 );
}
```

上記のように、数行の実行文を記述することでリレー操作するプログラムを作ることができます。

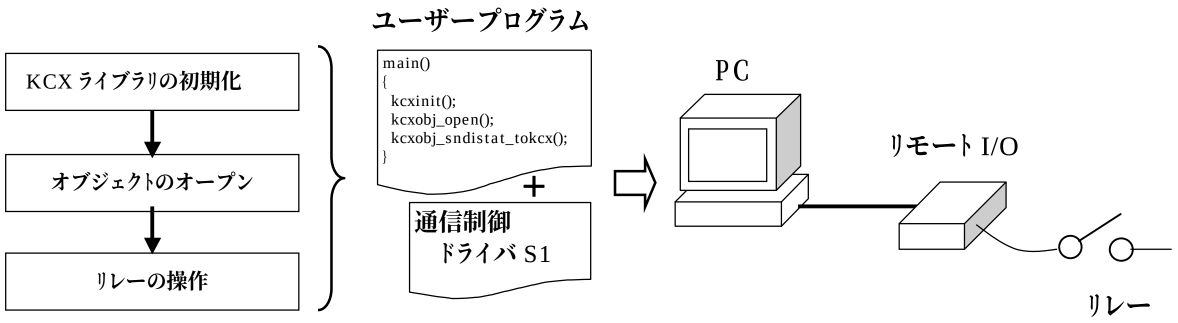
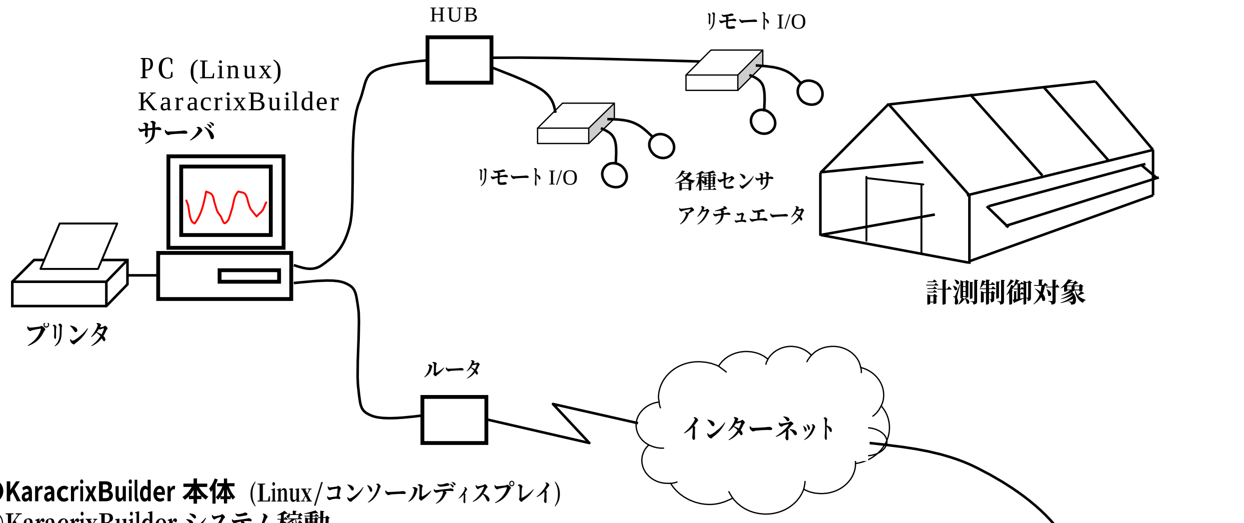


図 1.15.1 リレー操作プログラムの実行

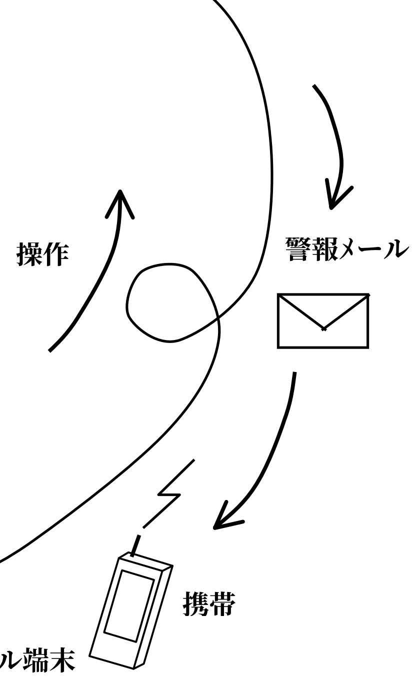
1.16　KaracrixBuilder＋リモート I/O 装置で実現できること

KaracrixBuilder とリモート I/O 装置を組み合わせることで以下のようなシステムを構築できます。



●KaracrixBuilder 本体 (Linux/コンソールディスプレイ)

○KaracrixBuilder システム稼動
○リモート I/O 装置との通信制御＆データ計測
○ディスプレーによるリアルタイム監視と操作
○計測データの定時記録（ファイル）、操作記録管理
○接点＆アナログ上下限警報発信、警報記録管理
○警報ブザー、警報ロック、積算カウント自動回復
○帳票印刷（製品版）
○必要な自動制御プログラムの作成＆実行



●KaracrixBuilder 遠隔操作 (WindowsPC(IE/Firefox)等/携帯(下記一部))

○ポイントオブジェクトデーター一覧表示＆操作
○監視モニタ/トレンドグラフ画面表示＆操作
○自動制御パラメータ編集
○計測記録データ(CSV)のダウンロード
○操作/警報履歴データ(CSV)のダウンロード
○帳票(PDF)のダウンロード＆印刷（製品版）
○警報メール受信/操作メール送信

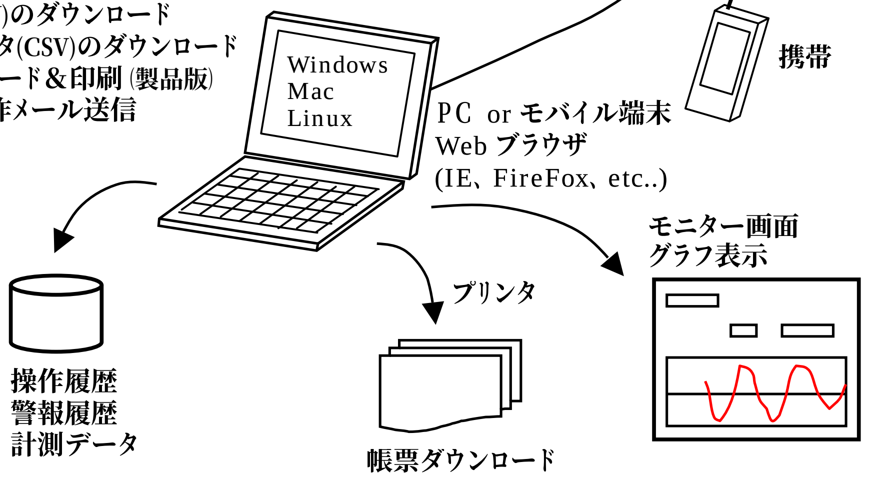


図 1.16.1 KaracrixBuilder で構築するシステム

付 録

●センサやアクチュエータをリモート I/O 装置に接続(実体配線)する手引き

センサやアクチュエータを、リモートI/O 装置に接続する場合の具体的な接続方法をいくつか紹介します。弊社製品の KaracriBoard-TK0040A (以下、TK0040A と省略) を例に、一般的に良く使われているセンサとアクチュエータのいくつかを取り上げますので参考にして下さい。
接点入力、リレー出力、アナログ入力のそれぞれ1つあるいは2つを、リモート I/O 装置のチャネル 1 あるいはチャネル 2 へつなげています。接続例では、TK0040A 用の専用簡易端子台を使用しました。
説明では、装置端子は信号名で表します。装置側内部の回路は表記しませんので、弊社HPのダウンロードページより取得できる「ディジタル・アナログ I/O 装置 KaracriBoard-TK0040A 取扱説明書」 の内部回路図を参考にして下さい。
接点回路、リレー回路には、非絶縁タイプと絶縁タイプがありますが、これは装置外部の電気ノイズが装置内部に入らないようにするかどうかの回路選択で、絶縁タイプの場合、ノイズによる誤動作に強くなる一方で電源を外部に用意する手間とコストが余分にかかります。工業用途でよく使われています。

表 1.17.1 KaracriBoard-TK0040A 入出力 Dsub コネクタピン配置

端子番号	信号名	機能	端子番号	信号名	機能
19	12V	供給電源バイパス出力			
18	12V	供給電源バイパス出力	37	G	グランド
17	C	1～6ch 接点入力コモン(COM)	36	G	グランド
16	DI-1	接点入力 1	35	DI-4	接点入力 4
15	DI-2	接点入力 2	34	DI-5	接点入力 5
14	DI-3	接点入力 3	33	DI-6	接点入力 6
13	DO-1c	リレー(実装時)出力 1c (TR-1 連動)	32	TR-1	トランジスタ出力 1 (DO-1 出力)
12	DO-1a	リレー(実装時)出力 1a (TR-1 連動)	31	G	グランド
11	DO-2c	リレー(実装時)出力 2c (TR-2 連動)	30	TR-2	トランジスタ出力 2 (DO-2 出力)
10	DO-2a	リレー(実装時)出力 2a (TR-2 連動)	29	G	グランド
9	DO-3c	リレー(実装時)出力 3c (TR-3 連動)	28	TR-3	トランジスタ出力 3 (DO-3 出力)
8	DO-3a	リレー(実装時)出力 3a (TR-3 連動)	27	DO-3b	リレー(実装時)出力 3b (TR-3 連動)
7	DO-4c	リレー(実装時)出力 4c (TR-4 連動)	26	TR-4	トランジスタ出力 4 (DO-4 出力)
6	DO-4a	リレー(実装時)出力 4a (TR-4 連動)	25	DO-4b	リレー(実装時)出力 4b (TR-4 連動)
5	AI-1	アナログ入力 1	24	PWM-1	PWM 出力 1
4	AI-2	アナログ入力 2	23	PWM-2	PWM 出力 2
3	AI-3	アナログ入力 3	22	AO-1	アナログ出力 1
2	AI-4	アナログ入力 4	21	AO-2	アナログ出力 2
1	5V	アナログ用安定化電源 5V 出力	20	G	アナログ・グランド

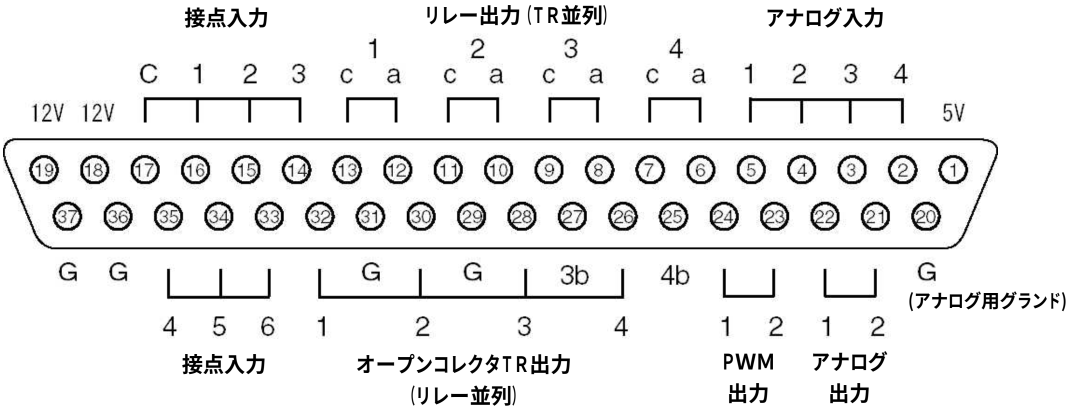


図 1.17.1 KaracriBoard-TK0040A 入出力コネクタピン配置(Dsub37 ピン)

12V	=	本機に供給している電源のバイパス出力 (バイパス回路にポリスイッチ(リセッタブルヒューズ)0.7A有り)
5V	=	アナログ用安定化電源(電圧精度1%、使用制限:50mA以下(ヒューズ無し注意))
G	=	グランド (マイナス電源に接地)
接点入力	=	無極性フォトカプラ接点入力
リレー出力1,2	=	ノーマルオープンa接点リレー,TR連動出力(リレー実装時)
リレー出力3,4	=	ノーマルca間オープンcbショートc接点リレー,TR連動出力(リレー実装時)
TR	=	オープンコレクタトランジスタ出力
AI	=	10ビット分解能アナログ入力 (増幅アンプ付)
AO	=	8ビット分解能アナログ出力
PWM	=	R/C用PWM出力用TTL出力

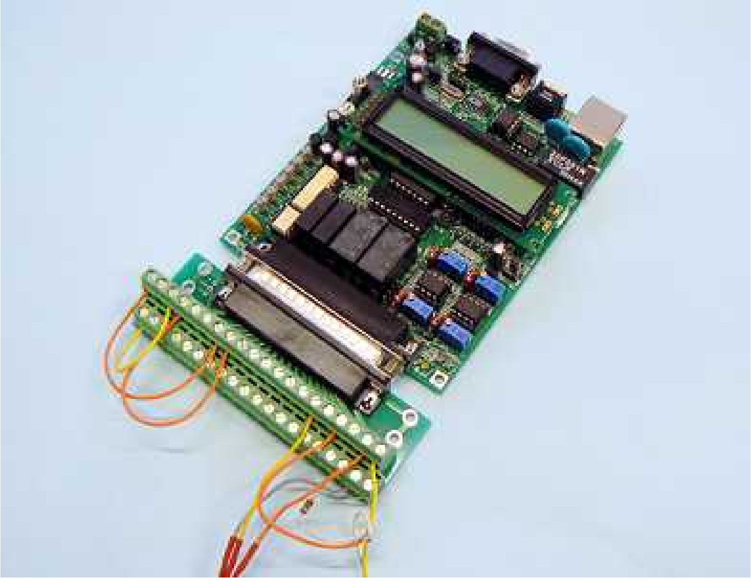


写真 1.17.1　TK0040A と簡易端子台
(KaracriBoard-TK0040A 本体に接続される簡易端子台 KB-CN-37A)

写真 1.17.1 は、KaracriBoard-TK0040A 本体と簡易端子台 (KB-CN-37A) を接続したところです。
下図 1.17.2 はこの簡易端子台のピン接続図で以降の説明で使用します。

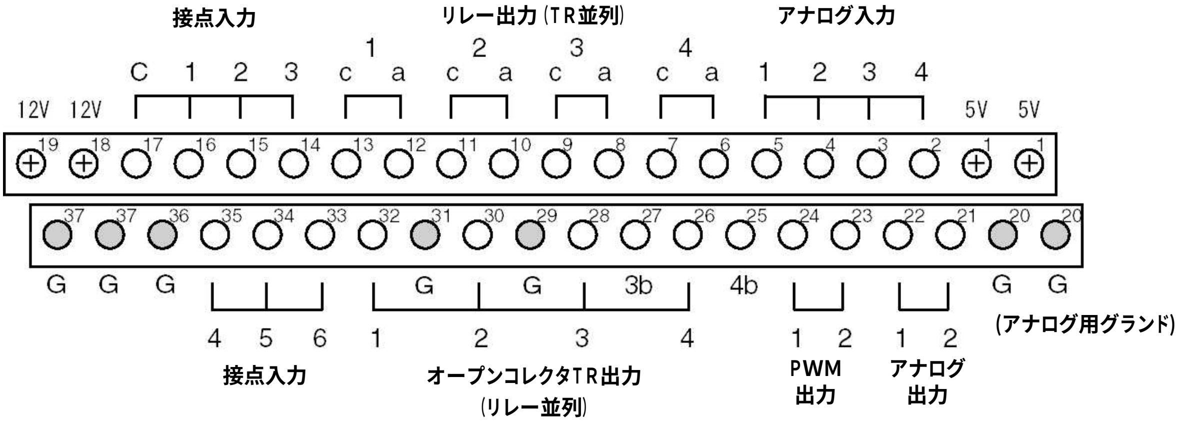
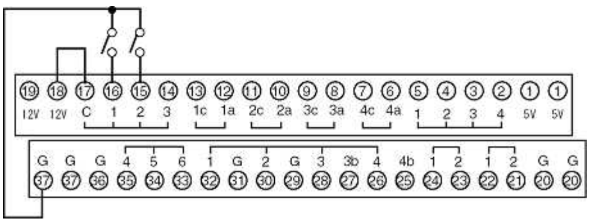
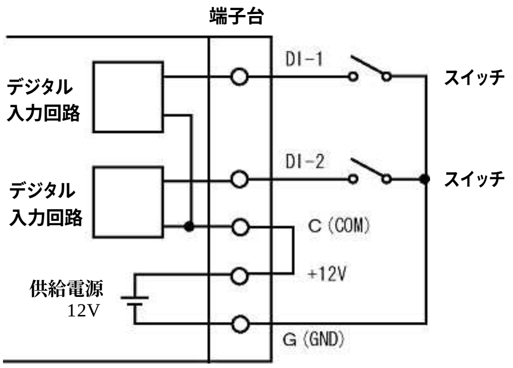


図 1.17.2 簡易端子台配置図

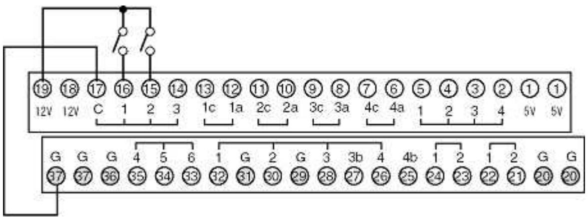
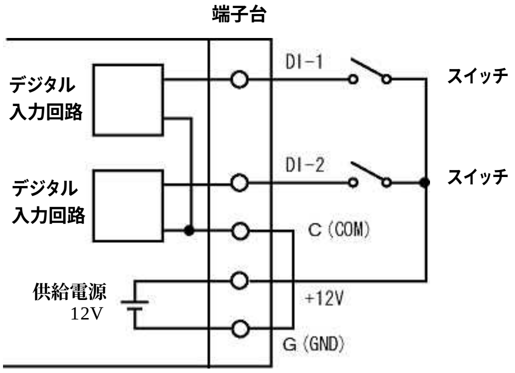
以下で、入出力端子台への配線の仕方について一覧にして解説しています。

A．接点入力 （スイッチ、フォトカプラトランジスタ等）

＜一般用途＞

<table>
<tr><th></th><th>配線図 (+電源･COM 接続時)</th><th>回路図 (+電源･COM 接続時)</th></tr>
<tr><td>非絶縁型</td><td>
</td><td>

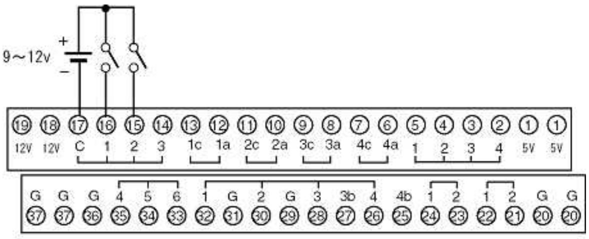
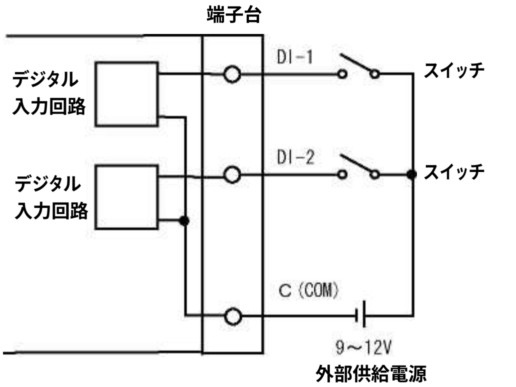
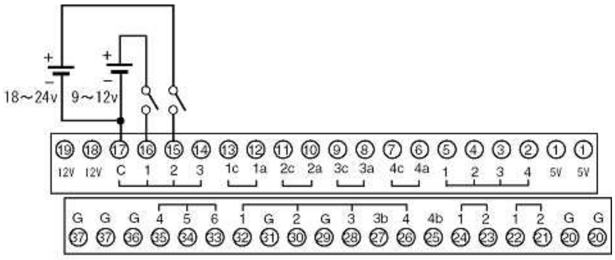
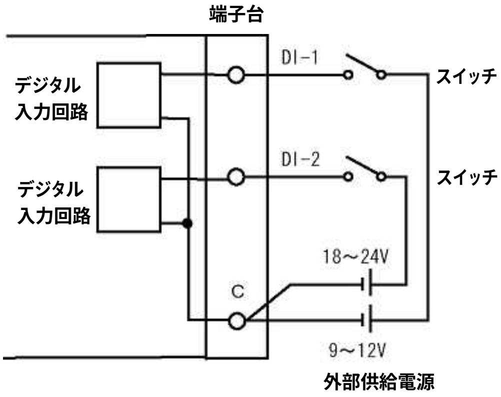
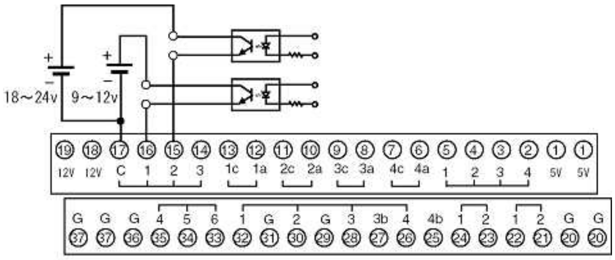
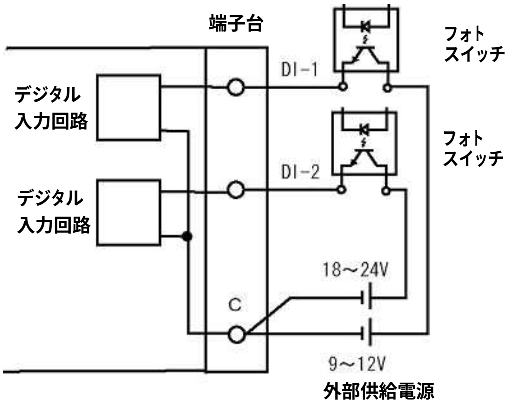
※COM と+12V の結線は内部電源使用を意味します</td></tr>
</table>

<table>
<tr><th></th><th>配線図 (グランド･COM 接続時)</th><th>回路図 (グランド･COM 接続時)</th></tr>
<tr><td>非絶縁型</td><td>
</td><td>

※スイッチと+12V の結線は内部電源使用を意味します</td></tr>
</table>

1. 装置内のデジタル入力回路に、外部取り付けスイッチより電流を DI 端子に流し込むように接続すると接点の ON/OFF が検出できるようになります。
2. TK0040A の場合、装置内のデジタル入力回路は無極性になっていますので電源(+-)の極性は自由です。
3. スイッチ 6 つまで同じような回路で拡張できます。

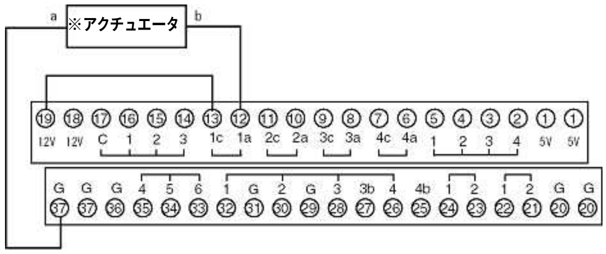
＜主に工業用途＞

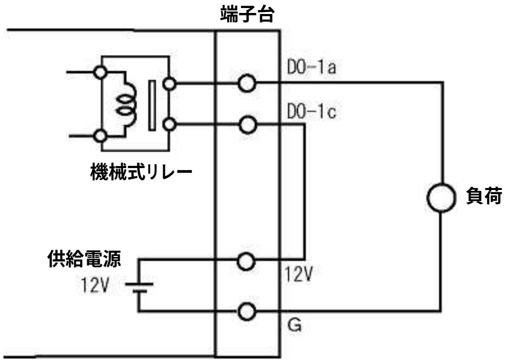
<table>
<tr><th></th><th>配線図</th><th>回路図</th></tr>
<tr><td>絶縁型（単電源）</td><td></td><td></td></tr>
<tr><td>絶縁型（複電源）</td><td></td><td></td></tr>
<tr><td>絶縁型（フォトカプラ入力）</td><td></td><td></td></tr>
</table>

1. TK0040A の場合、装置内のデジタル入力回路は無極性になっていますので電源(+-)の極性は自由です。但し、外部電源等複数使用する場合には電源間の極性にご注意ください。（絶縁型フォトカプラ入力の場合は、仕様（エミッタ、コレクタ）により外部電源の極性にご注意下さい）

B. オンボードリレー実装時のリレー出力 (電球、ブザー、ソレノイド、バルブ等)

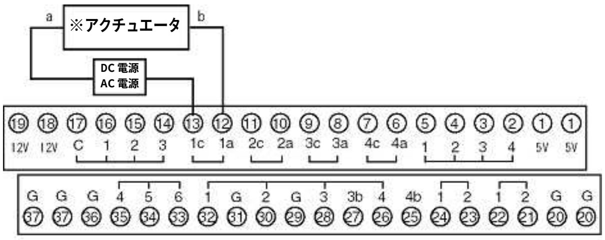
<一般用途>

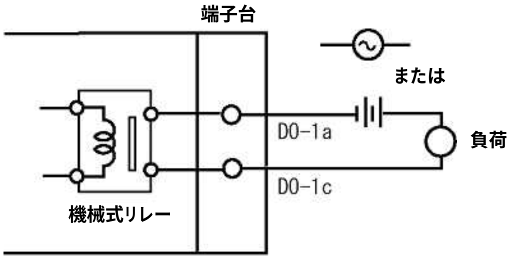
<table>
<tr><th></th><th>配線図</th><th>回路図</th></tr>
<tr><td>非絶縁型</td><td>

※アクチュエータ:分類表1参照</td><td>
</td></tr>
</table>

1. 上図電気負荷は、リレーの定格以内でお使いください。また、TK0040A から電源を得る場合、装置内に自己復帰型の電流ヒューズ(0.7A)が実装されていますのでこの定格以内でご使用ください。
2. 分類表１の①LED の抵抗Rは、1K～47KΩの範囲で光量を見ながら調節してください。
3. 分類表１の⑦⑧は電子知識がある方のみご使用ください。

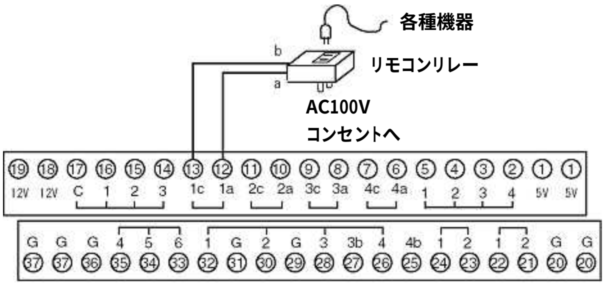
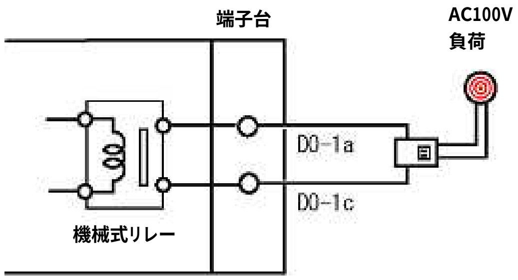
<主に工業用途>

<table>
<tr><th></th><th>配線図</th><th>回路図</th></tr>
<tr><td>絶縁型</td><td>

※アクチュエータ:分類表1参照</td><td>
</td></tr>
</table>

1. 外部電源には、AC、DC とも装置内リレー定格まで使えます。
2. 分類表１の見方は、外部電源の種類別に使用可不可を電気的に判断しご使用ください。

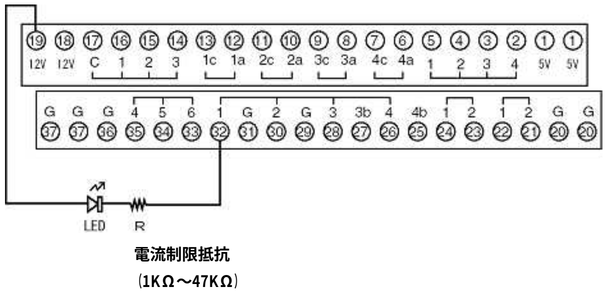
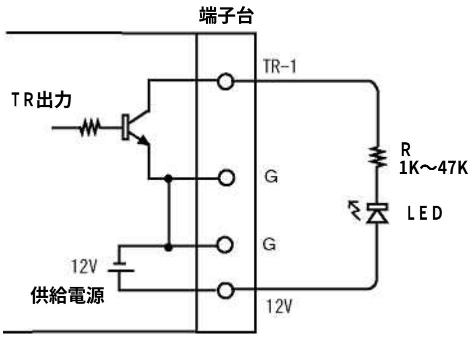
＜家庭用途＞

	配線図	回路図
リモコンリレー(絶縁型)		

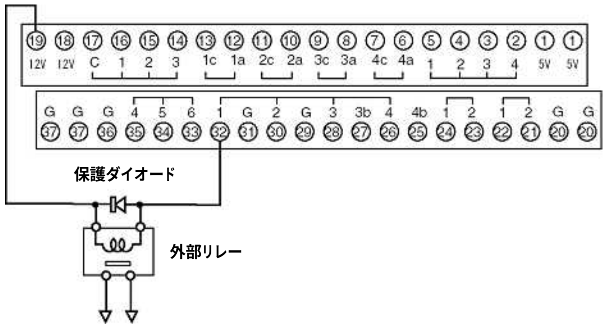
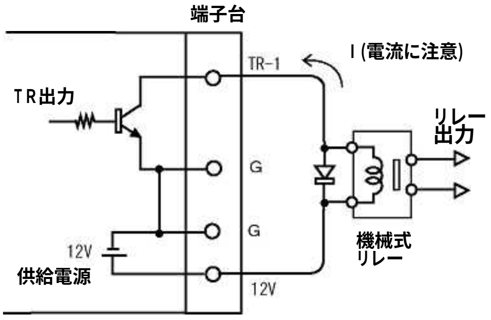
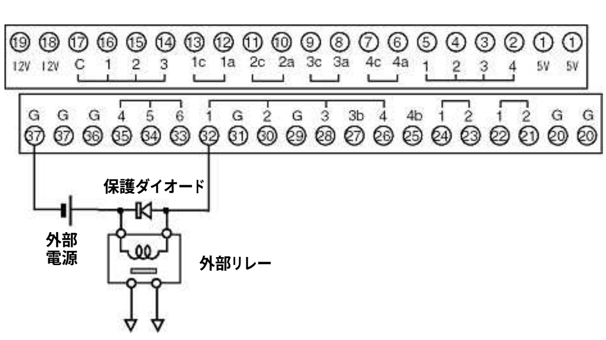
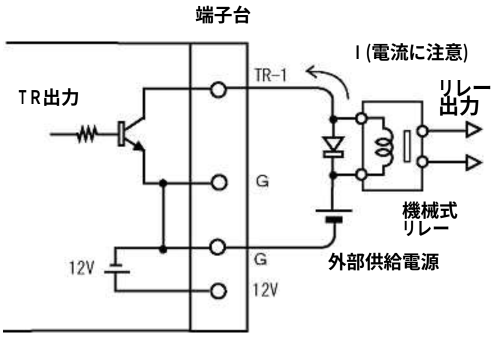
1. 上記リモコンリレーは、小さな接点出力で100Vを制御するものです。(一般家庭用に販売されています)
2. 100Vの各種機器が使えるようになります。以下に例を上げてみます。
   a) クリスマスツリー
   b) 照明器具
   c) モーター
   d) 換気扇
   e) ポンプ
   f) コンロ
   g) 扇風機
   h) ラジオ
   I) 室内スイッチ
   j) ファン
   k) その他各種100V装置

C．トランジスタ出力（TR出力）

TR出力は、リレー出力と並列に接続されており同時に使用することが可能です。代表的な使用例を以下に示します。

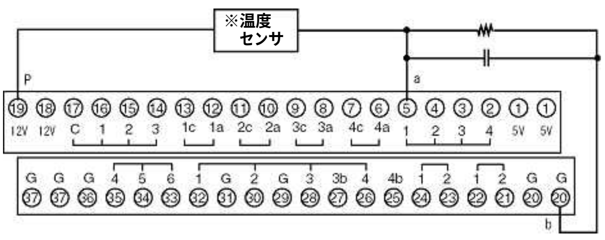
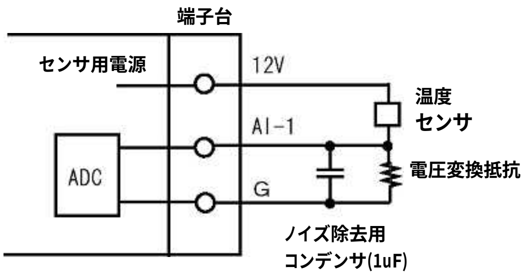
	配線図	回路図
内部電源使用		

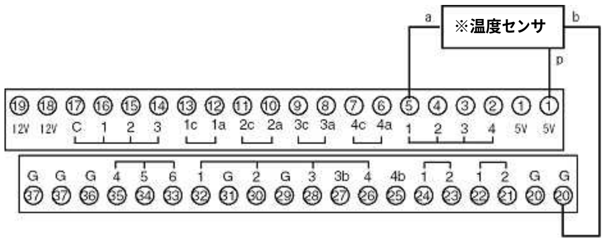
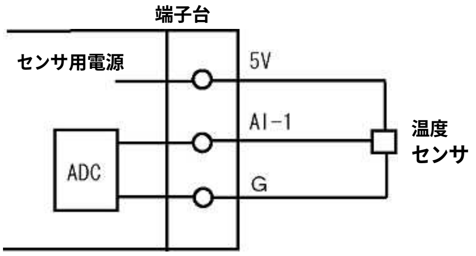
1. TR出力で供給電源を使用してLEDをドライブする場合の回路図です。LED の電流制限抵抗Rは、1K〜47KΩの範囲で光量を見ながら調節してください。

	配線図	回路図
内部電源使用		
外部電源使用	 グランド(G)は、37、36、31、29、20番内部接続され共通です。	

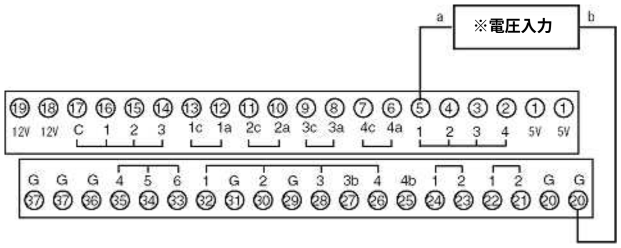
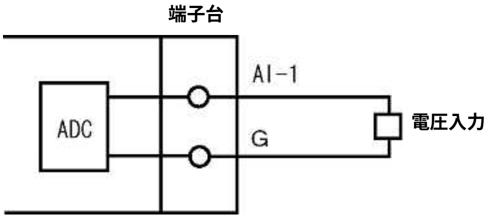
1. TR出力で機械式リレーをドライブする場合、TRの最大駆動電流と外部供給電源を使用する場合には駆動電圧が定格を超えないように注意して下さい。

D. アナログ入力 (各種センサ等)

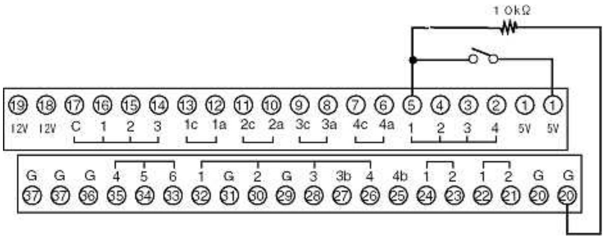
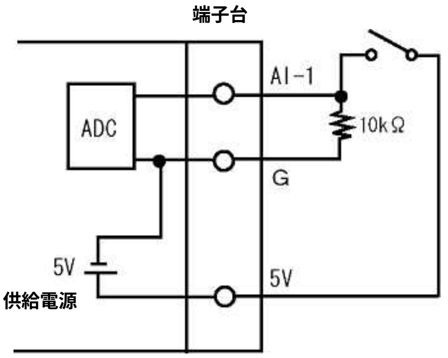
	配線図	回路図
温度センサ（電流型）	 ※各種センサ：分類表2参照	

	配線図	回路図
温度センサ（電圧型）	 ※各種センサ：分類表2参照	

1. 各種センサの出力が+5.2V(TK0040A時)以内になるようにご使用ください。
2. 電流型センサへの信号線は十分長く伸ばせますが、電圧型の場合は配線抵抗の影響があります。
3. センサには固体差がありますので、精密計測する場合には、この特性を観測し計測データを校正する必要があります。

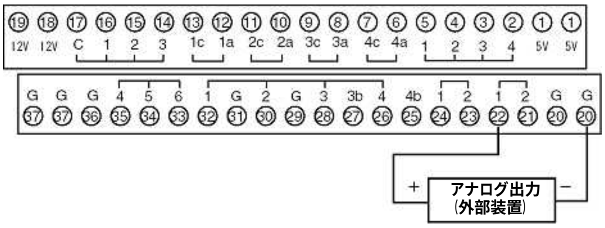
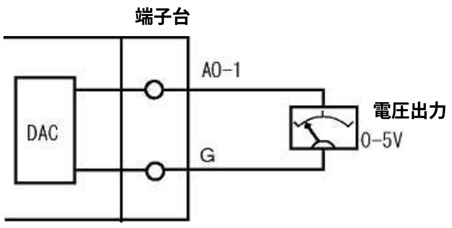
	配線図	回路図
電圧入力	 ※電圧入力：分類表3参照	

1. AI端子には、+5.2V(TK0040A時)以内になるような電圧を与えてください。

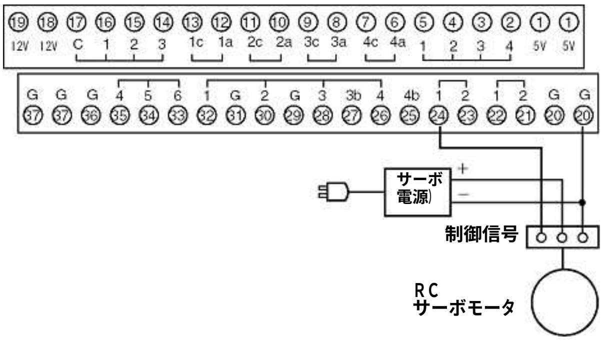
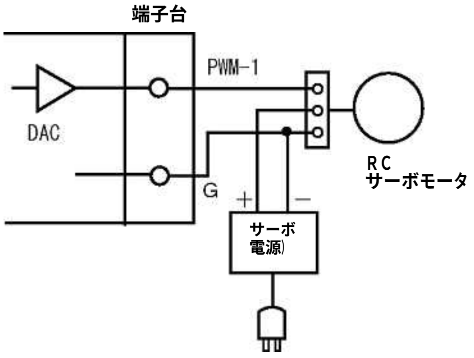
	配線図	回路図
スイッチ入力		 (スイッチ OFF→0V、ON→5V)

1. アナログ入力で、スイッチの状態を検出する例を示します。（スイッチを通して電圧を検出させる応用）

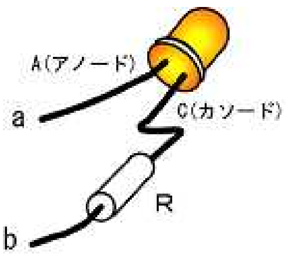
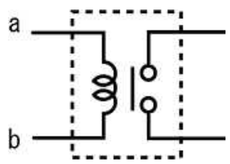
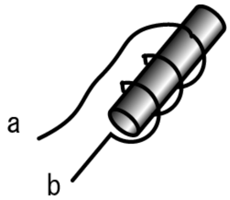
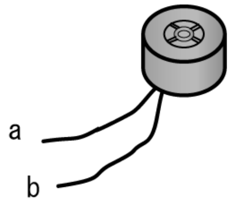
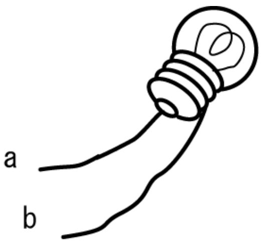
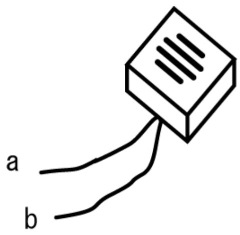
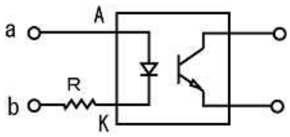
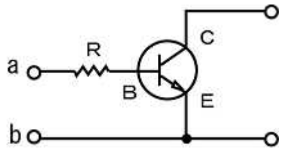
E. アナログ出力 （各種装置電圧制御等）

<table>
<tr><th></th><th>配線図</th><th>回路図</th></tr>
<tr><td>各種外部装置</td><td>
</td><td>
</td></tr>
</table>

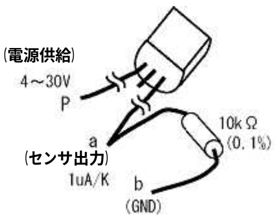
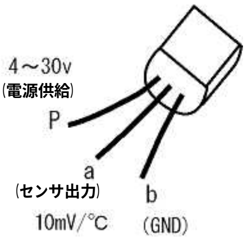
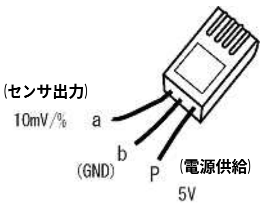
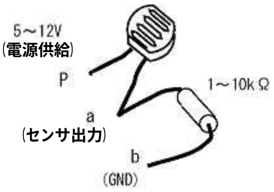
F. PWM出力 （RCサーボモーター用）

<table>
<tr><th></th><th>配線図</th><th>回路図</th></tr>
<tr><td>サーボモーター</td><td>
</td><td>
</td></tr>
</table>

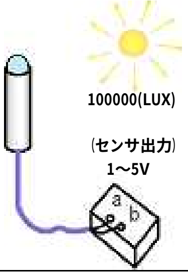
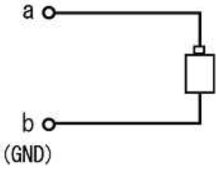
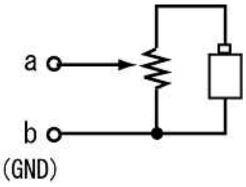
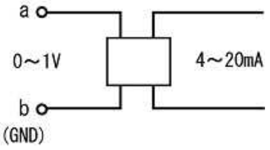
分類表1 （各種アクチュエータ）

1) LED	2) パワーリレー	3) ソレノイド	4) ブザー
 ※抵抗 R はアノード側でも可		 ※a-b 間には CR が必要です	
5) 豆電球	6) ドアチャイム	7) フォトカプラ	8) トランジスタ
			

分類表２ （各種センサ）

温度(AD592)	温度(LM35D)	湿度(CHS-UGS)	光(Cds)
電流型温度センサ  ※電流型センサは、計測ポイントまでの配線を長くできますが、高精度の抵抗で電圧変換を行う必要があります。	電圧型温度センサ 	電圧型湿度センサ 	電圧型光センサ  ※Cds の抵抗変化を電圧の変化として検出します。

分類表３ （電圧入力）

光（フォトダイオード）	固定電圧	抵抗分割降圧	電流-電圧 (絶縁変換器を使用)
電圧型照度センサ 			

<table>
<tr><td colspan="2">○製品紹介
弊社から販売しているリモートI/O装置(LAN対応)には、「KaracriBoard-TK0040A」「NetBoxシリーズ」等がご利用になれます。</td></tr>
<tr><td>KaracriBoard-TK0040A
 </td><td>1. フォトカプラ接点入力×6
2. トランジスタ(機械式リレーオプション)出力×4
3. +5V アナログ(10bit)入力×4
4. +5V アナログ(8bit)出力×2
5. RC/PWM 出力×3
6. 12V 電源</td></tr>
<tr><td>NetBox-E100-BK1682A
 </td><td>1. フォトカプラ接点入力×16
2. 機械式リレー(脱着可能)出力×4
3. トランジスタ出力×2
4. 24V 電源 / 筐体 DIN レール取り付け可</td></tr>
<tr><td>NetBox-E100-AK0822A
 </td><td>1. +5V,+10V,±5V,±10V アナログ(12bit)入力×8
2. +5V アナログ(8bit)出力×2
3. トランジスタ出力×2
4. 24V 電源 / 筐体 DIN レール取り付け可</td></tr>
<tr><td>NetBox-E200-AK0620A
 </td><td>1. 絶縁/非絶縁アナログ(16bit)入力×12
1.2v,2.4v,6v,12v,±1.2v,±2.4v,±6v,±12v,
0-20mA,4-20mA (0.1%精度シャント抵抗内臓)
2. アナログ(12bit)出力×2
+5V,+10V,±5V,±10V,0-20mA,4-20mA
3. フォトカプラ接点入力×2
4. トランジスタ出力×2
5. 24V 電源 / 筐体 DIN レール取り付け可</td></tr>
</table>

株式会社エスアイ創房　http://www.karacrix.jp/